

# SQL Link for MS Excel 1.

## User Manual

1st Dept: 20		
2nd Dept: 30		
select * from dept		
where deptno between &d2 and &d3		
order by deptno desc		
<b>DEPTNO</b>	<b>DNAME</b>	<b>LOC</b>
30	SALES	CHICAGO
20	RESEARCH	DALLAS

update emp
set sal = sal + 50
14 rows affected
delete emp
where deptno = 20
1 rows affected

<b>DEPT</b>	<b>ID</b>	<b>Dname</b>	<b>Loc</b>
Expr deptno			
Include			
Order By			
Group By			
Having			
where >=			
AND <=			
OR			
SELECT deptno ID, dname			
Dept, loc Loc FROM DEPT			
WHERE ( (deptno>=20)) AND			
( (deptno<=30))			
<b>ID</b>	<b>DEPT</b>	<b>LOC</b>	
20	RESEARCH	DALLAS	
30	SALES	CHICAGO	

Developed and Written by: M. Armaghan Saqib

© 1998-1999 by M. Armaghan Saqib & Associates

Email:

[armaghan@geocities.com](mailto:armaghan@geocities.com)

Web site:

<http://www.geocities.com/WallStreet/Floor/8484/>

<http://armaghan.webjump.com>



## Table of Contents

<b>1. INTRODUCTION.....</b>	<b>5</b>
<b>2. INSTALLATION AND GETTING STARTED.....</b>	<b>6</b>
1. WHAT IS INCLUDED IN THE SQL LINK PACKAGE .....	6
2. INSTALLATION REQUIREMENTS.....	6
3. INSTALLING SQL LINK AS AN ADD-IN IN MICROSOFT EXCEL.....	7
4. OPENING SQL LINK ONLY WHEN REQUIRED .....	7
5. CREATING NEW EXCEL WORKBOOKS FOR USE WITH SQL LINK .....	7
6. USING EXISTING EXCEL WORKBOOKS FOR USE WITH SQL LINK .....	8
<b>3. FEATURES.....</b>	<b>9</b>
1. ORACLE DATABASE FEATURES .....	9
2. ODBC DATABASE CONNECTIVITY FEATURES .....	9
3. OTHERS FEATURES.....	9
<b>4. USING SELECT QUERIES .....</b>	<b>10</b>
1. CONNECTING TO DATABASE .....	10
2. EXECUTING SQL FROM WORKSHEET CELLS .....	10
3. DEFINING USER PARAMETERS USING SQL PLUS SYNTAX.....	10
4. USING AUTOMATIC GROUPS AND FILTERS ON QUERIES.....	11
5. USING DICTIONARY MENU.....	12
<b>5. INSERTS AND UPDATES .....</b>	<b>13</b>
1. PREPARING FOR INSERT .....	13
2. INSERTING ROWS AND HANDLING DATABASE ERRORS.....	13
3. UPDATING DATA AND HANDLING DATABASE ERRORS .....	14
4. USING PRIMARY KEYS/ROWIDS FOR UPDATE .....	15
5. COMMIT AND ROLLBACK OF INSERTS AND UPDATES.....	15
<b>6. EXECUTING PL/SQL.....</b>	<b>16</b>
1. EXECUTING INSERT, UPDATE, DELETE, CREATE ETC. ....	16
2. EXECUTING PL/SQL .....	16
3. COMMIT AND ROLLBACK .....	16
<b>7. QUERY BY EXAMPLE (QBE).....</b>	<b>17</b>
1. CREATING QBE GRID .....	17
2. SELECTING QUERY COLUMNS AND SPECIFYING CRITERIA .....	18
3. USING COLUMN EXPRESSIONS .....	18
4. USE GROUP BY AND HAVING ROWS.....	18
5. CUSTOMIZING QBE GRID .....	18
6. SPECIFYING QBE OPTIONS .....	18
<b>8. SQL LINK OTHER FEATURES.....</b>	<b>19</b>
1. GENERATING SIMPLE SQL STATEMENTS .....	19
2. USING KEY BUILDERS.....	19

3. USING REPORT BUILDERS .....	19
<b>9. BUILDING SOLUTIONS WITH MACROS .....</b>	<b>20</b>
1. AUTO_CONNECT AND AUTO_EXECUTE WHEN YOU OPEN A WORKSHEET .....	20
2. CALLING SQL LINK PROCEDURES/MACROS DIRECTLY .....	20
3. USING SENDKEYS ACTION TO EXECUTE SQL LINK COMMANDS .....	20
<b>A. ODBC INTERFACE.....</b>	<b>23</b>
1. EXECUTING SELECT QUERIES.....	23
2. SPECIFYING USER PARAMETERS .....	24
3. SPECIFYING OTHER QUERY OPTIONS .....	24
4. MODIFYING QUERIES USING MS QUERY BUILDER.....	25
5. CHANGING SQL CREATED BY QUERY BUILDER.....	25
6. LIMITATIONS OF ODBC INTERFACE IN SQL LINK 2.0 .....	25
<b>B. SQL LINK MENU AND MACROS REFERENCE.....</b>	<b>26</b>
7. EXECUTE.....	26
8. CONNECT .....	27
9. DICTIONARY.....	27
10. QUERY BY EXAMPLE.....	28
11. DATA .....	28
12. GENERATE .....	30
13. KEY BUILDERS.....	30
14. REPORT BUILDERS .....	31
15. ORACLE LINK .....	31
16. ODBC .....	32
17. ABOUT .....	32
<b>C. TROUBLE SHOOTING AND TECHNICAL SUPPORT .....</b>	<b>33</b>
<b>D. LICENSE AND DISCLAIMER.....</b>	<b>34</b>

# 1. Introduction

SQL Link for MS Excel is a tool, which allows you to combine the flexibility of Microsoft Excel with power of databases.

It provides native access to Oracle database using Oracle Objects for OLE for high-speed data access.

You can compare it with Oracle SQL Plus. SQL Plus is a very powerful tool, which allows you to extract data from Oracle database tables. You can also INSERT, UPDATE and DELETE with SQL Plus.

SQL Link for MS Excel does the same but within the environment of MS Excel. This capability gives you the tremendous power of MS Excel to apply on Oracle data.

Current version of SQL Link also gives you the power to access any ODBC database using its ODBC connectivity.

SQL Link for MS Excel provides an excellent environment for ad hoc reporting. With its batch execution facility combined with SQL Plus like “&” variables, auto-connect and auto-execute on startup allows you to create presentation quality reports complete with user-supplied parameters for reports.

## 2. Installation and Getting Started

Install Oracle Object for OLE (supplied by Oracle Corporation with almost all distributions of Oracle Database).

Copy the worksheet in whatever directly you want to use.

Open the worksheet and enter your user name, password and database name on Options worksheet.

Start learning its functionality using sample SQL statements provided. For this you must be connected as user 'scott'.

When you are familiar with the tool, delete the samples and start using your own SQL statements.

**WARNING:** In the current version of SQL Link, you cannot UNDO any changes made by SQL Link commands. So be a little more careful when using SQL Link commands on critical data. And remember to keep backups always.

### 1. What is included in the SQL Link Package

SQL Link is available as a self-extracting executable file **sqllink.exe** on the web site. Following is the description of the files included in the self-extractor.

Sqllink150.exe contains following three files:

- |                |   |
|----------------|---|
| 1. sqllink.xla | SQL Link addin for MS Excel   |
| 2. samples.xls | Demonstrations of SQL Link features. Can be used as a starting point to make your own worksheets. |
| 3. manual.pdf  | SQL Link user manual with smaller page size optimized for on-screen viewing                       |

### 2. Installation Requirements

SQL Link requires MS Excel 97 or higher. In addition it requires the following components to operate as required:

Oracle Objects for OLE (oo4o): This is required for native connectivity to Oracle. OO4O comes on almost all installation CDs of Oracle database. This is not required if you will only use ODBC functionality. You can download OO4O trial version from Oracle web site. Here are the links:

<http://www.oracle.com/products/trial/>

<ftp://ftp.oracle.com/pub/www/oo4o/>

Data Access Objects (DAO):	If you want to use ODBC functionality, you must install MS Excel with DAO 3.5 or higher
Microsoft Query (Optional):	This optional component of Microsoft Excel allows you to edit ODBC queries in a user-friendly graphical environment.

### 3. Installing SQL Link as an Add-in in Microsoft Excel

To install SQL Link as an Add-in to MS Excel, follow this step-by-step procedure:

1. Copy sqllink.xla in any folder (e.g. c:\sqllink\ ) on your hard drive.
2. Select Tools:Add-Ins from Excel menu.  
A list of installed add-ins will be displayed along with Ok Cancel Browse buttons. There will be a check mark to the left of each active add-in.
3. Click Browse and a file open dialog will appear.
4. Select sqllink.xla using this dialog.  
It will be installed in the MS Excel as an Add-in and will open automatically whenever you will open the worksheet.

### 4. Opening SQL Link Only When Required

It might be required by some of the users that SQL Link does not automatically opens whenever they start MS Excel.

This can be achieved by putting a default auto\_open macro in your worksheet. Use SQL:SQL Link:Auto\_Open Macro to insert such a macro.

Now whenever you open worksheet containing this macro, sqllink.xla will automatically open. Samples.xls can be good starting point for this purpose.

(Note: Any reference to sqllink.xla in a work sheet will open sqllink.xla automatically)

### 5. Creating New Excel Workbooks for Use with SQL Link

SQL Link operation requires that two sheets (Options and SQL\_Batch) be present in the working workbook with specific information.

After you create blank workbooks using File – New command, you should also create these two sheets in that workbook.

You can create both these sheets through SQL Link by “SQL : SQL Link : Insert Options Sheets” command.

## 6. Using Existing Excel Workbooks for Use with SQL Link

Similarly you need these two sheets (Options and SQL\_Batch) in all existing worksheets which you intend to use with SQL Link.

Use the same procedure as described above to achieve this.



## 3. Features

### 1. Oracle Database Features

Execute SQL and PL/SQL from cells

Execute SQL from cell comments (mimics XL formulas)

Execute more than one SQL statements at once from a batch SQL sheet

Automatically create Groups and subtotals after SQL execution

Auto-connect to database on opening of worksheet

Auto-execute SQL statements on opening of worksheet

Extract data from any number of users on any number of databases all in one go.

Easily browse database tables structure and data with menu commands.

User SQL Plus like “&” variables to create parameters for all your SQL statements.

### 2. ODBC Database Connectivity Features

Execute SQL from any worksheet cell.

Specify user parameters using ? in the SQL.

Specify result set properties (range name, parameter names and values etc) directly by right clicking on the result set.

Refresh on open can be specified to automatically refresh the data when worksheet opens.

ODBC connectivity is a new feature in SQL Link. It uses Microsoft DAO technology. Currently you cannot do INSERTs or UPDATEs to ODBC databases but future versions of SQL Link will allow you to do so.

### 3. Others Features

Insert a SUM function for currently selected cells.

Insert a VLOOKUP function for a data range using Range Names

Insert a VLOOKUP function for a data range using Selected Range

## 4. Using SELECT Queries

### 1. Connecting to Database

You can connect to database using Connect:Connect User command and specifying user name, password and database alias.

You can also store user information in “Options” sheet and connect to database using Connect:Connect Default User command.

You can also connect automatically to database whenever you open your worksheet. To do this you must do the following:

1. Specify user and database information in “Options” sheet.
2. Set Auto Connect? to TRUE in “Options” sheet.
3. Finally create a default auto\_open macro in your sheet using SQL Link : Auto\_Open Macro command.

### 2. Executing SQL from worksheet cells

You can execute an SQL statement from any worksheet range. Simply enter your SQL in as many worksheet rows as you like.

To execute this sql, position your cursor in any cell and select SQL:Execute:SQL in Cell.

A1/								
	A	B	C	D	E	F	G	H
1								
2	SELECT * FROM EMP							
3	WHERE SAL BETWEEN 1000 AND 2000							
4	ORDER BY HIREDATE							
5								
6	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPT
7	7499	ALLEN	SALESMAN	7698	2/20/81	1600	300	
8	7521	WARD	SALESMAN	7698	2/22/81	1250	500	
9	7844	TURNER	SALESMAN	7698	9/8/81	1500	1/0/00	
10	7654	MARTIN	SALESMAN	7698	9/28/81	1250	10/31/03	
11	7934	MILLER	CLERK	7782	1/23/82	1300		
12	7876	ADAMS	CLERK	7788	5/23/87	1100		

### 3. Defining User Parameters using SQL Plus Syntax

You can specify parameters SQL Plus like & variables. The following example stores highest and lowest salary in two worksheet cells E2 and E3. These cells are then referenced in SQL using &E2 and &E3.

If you are specifying parameters for varchar2 columns, you must enclose parameters in single quotes e.g. '&E2'.

	C	D	E	F	G	H	I	J	
	Lowest Salary		1000						
	Highest Salary		2000						
	SELECT * FROM emp WHERE sal BETWEEN &e2 AND &e3								
	<b>EMPNO</b>	<b>ENAME</b>	<b>JOB</b>	<b>MGR</b>	<b>HIREDATE</b>	<b>SAL</b>	<b>COMM</b>	<b>DEPTNO</b>	
	7499	ALLEN	SALESMAN	7698	2/20/81	1600	300	30	
	7521	WARD	SALESMAN	7698	2/22/81	1250	500	30	
	7654	MARTIN	SALESMAN	7698	9/28/81	1250	1400	30	
	7844	TURNER	SALESMAN	7698	9/8/81	1500	0	30	
	7876	ADAMS	CLERK	7788	5/23/87	1100		20	
	7934	MILLER	CLERK	7782	1/23/82	1300		10	

#### 4. Using Automatic Groups and Filters on Queries

You can group your query results and calculate subtotals for each group. First column of your query results is used to group the results and last column is used to calculate subtotals.

To calculate groups and subtotals set "Calculate Subtotals?" to TRUE in "Options" sheet.

Similarly you can further filter your data retrieved from Oracle using XL filters. If you set "Auto Filter?" to TRUE then SQL Link will insert the filter buttons in your query results.

Note: Due to current Microsoft XL limitations, you can use filter buttons with only one data range on a single worksheet.

A21

1	2	3	A	B	C	D	E
1							
2			select deptno, empno, ename, sal from emp				
3			where deptno < 30 order by deptno, empno				
4							
5			DEPTNO	EMPNO	ENAME	SAL	
6			10	7782	CLARK	2450	
7			10	7839	KING	5000	
8			10	7934	MILLER	1300	
9			10 Total			8750	
10			20	7369	SMITH	800	
11			20	7566	JONES	2975	
12			20	7788	SCOTT	3000	
13			20	7876	ADAMS	1100	
14			20	7902	FORD	3000	
15			20 Total			10875	
16			Grand Total			19625	
17							

## 5. Using DICTIONARY menu

Using Dictionary menu you can easily navigate through the tables, structures and data in your database. These simple commands do now require you to type any SQL to browse your database.

Please see the demo1.scm for the demonstration of this feature.

## 5. INSERTs and UPDATEs

### 1. Preparing for INSERT

You can insert data into your Oracle database directly from your worksheets. Here is the procedure.

1. Enter an SQL statement on your worksheet.
2. From SQL Link menu select Data>Create Insert Rows Area  
SQL Link will bring up the table structure along with column types and sizes. This information helps you to enter data for INSERTing in the database.
3. You can now enter data on your worksheet.

A9			
	A	B	C
1			
2			
3	select * from dept		
4			
5	DEPTNO	DNAME	LOC
6	number	varchar2	varchar2
7	not null	null	null
8	2	14	13
9			
10			
11			

### 2. Inserting Rows and Handling Database Errors

1. To insert into database, select the SQL statement and give the command Data:Insert Rows. Rows will be inserted in the database but not committed yet.
2. To make your INSERTs permanent, give the command Data:Commit Session.
3. To undo your INSERTs, give the command Data:Rollback Session.
4. Error handling: SQL Link will display the database errors encountered during INSERTs in a column right the last data column.



#### **4. Using Primary Keys/ROWIDs for UPDATE**

These features are currently not implemented. Future versions will allow you to use Primary keys or ROWIDs to update your data.

Please visit your web site regularly to find updated version of SQL Link.

#### **5. COMMIT and ROLLBACK of INSERTs and UPDATEs**

Changes are not made permanent to database unless you commit them using Data:Commit Session command.

You can also undo any changes made to database using Data:Rollback Session command.

## **6. Executing PL/SQL**

1. Executing INSERT, UPDATE, DELETE, CREATE etc.
2. Executing PL/SQL
3. COMMIT and ROLLBACK



## 7. Query by Example (QBE)

SQL Link provides a QBE interface to construct SELECT queries. With the help of QBE grid, even novice users can create very complex SELECT statements.

QBE is also helpful for experts. It saves them typing a lot of column names, avoiding syntax errors etc.

A working knowledge of SQL is recommended to make full use of QBE feature.

	A	B	C	D	E	F	G	H	I
1	EMP								
2		<b>empno</b>	<b>ename</b>	<b>job</b>	<b>mgr</b>	<b>hiredate</b>	<b>sal</b>	<b>comm</b>	<b>deptno</b>
3	Expr	<i>empno</i>	<i>ename</i>	<i>job</i>	<i>mgr</i>	<i>hiredate</i>	<i>sal * 2</i>	<i>comm</i>	<i>deptno</i>
4	Include	.	.	.	.	.	.	.	.
5	Order By								<b>a</b>
6	Group By								
7	Having								
8	Where						>=1000		
9	AND						<=2000		
10	OR								
11	OR								
12	OR								
13									
14	SELECT empno, ename, job, hiredate, sal * 2 sal, comm, deptno FROM EMP WHERE ( (sal * 2>=1000)) AND ( (sal * 2<=2000)) ORDER BY deptno								
15									
16	<b>EMPNO</b>	<b>ENAME</b>	<b>JOB</b>	<b>HIREDATE</b>	<b>SAL</b>	<b>COMM</b>	<b>DEPTNO</b>		
17	7369	SMITH	CLERK	12/17/80	800		20		
18	7900	JAMES	CLERK	12/3/81	950		30		

### 1. Creating QBE Grid

To use QBE, type name of the table in a worksheet cell. While selecting this cell, give the command Query by Example : Build QBE Grid. A grid containing column names of your tables along with other information will be created (as show in the above screen output)

## 2. Selecting Query Columns and Specifying Criteria

## 3. Using Column Expressions

## 4. Use GROUP BY and HAVING rows

	B	C	D	E	F	G	H	I	J
3	EMP								
4		empno	ename	job	mgr	hiredate	dept_salary	comm	deptno
5	Expr	empno	ename	job	mgr	hiredate	sum(sal)	comm	deptno
6	Include						.		.
7	Order By								a
8	Group By								.
9	Having						<10000		
10	Where								
11	OR								
12	OR								
13	OR								
14	OR								
15									
16	SELECT sum(sal) dept_salary, deptno FROM EMP GROUP BY deptno HAVING (sum(sal)<10000) ORDER BY deptno								
17									
18	DEPT_SALARY	DEPTNO							
19	8750	10							
20	9400	30							

## 5. Customizing QBE Grid

You can delete any columns safely from QBE grid which you do not want in your select statement. Similarly you can arrange columns by moving columns back and forth thus bringing the columns in the order you want.

## 6. Specifying QBE Options

You can specify the following options on Options sheet to control the behavior of the QBE functionality.

	<b>QBE options:</b>	
	Default QBE WHERE Rows:	5
	Insert SQL on New Sheet?	FALSE
	Execute SQL?	TRUE
	Reuse QBE Sheet?	TRUE

## **8. SQL Link Other Features**

1. Generating Simple SQL Statements
2. Using Key Builders
3. Using Report Builders

## 9. Building Solutions with Macros

### 1. AUTO\_CONNECT and AUTO\_EXECUTE when you open a worksheet

You can connect to database automatically when you open your worksheet. Similarly you can automatically execute SQL statements on SQL\_Batch sheet when you open your worksheet.

To activate these startup settings you need to do two things:

1. Set Auto Connect? And Auto Execute? To TRUE in Options sheet
2. Create a default auto\_open macro in your worksheet using SQL:SQL Link:Auto\_Open Macro. SQL Link will insert the following macro in your current worksheet.

```
Sub Auto_Open()  
    Application.Run "sqllink.xla!Auto_Connect"  
    Application.Run "sqllink.xla!Auto_Execute"  
End Sub
```

### 2. Calling SQL Link Procedures/Macros Directly

You can also call SQL Link commands directly from your macros. Macro name for each SQL Link command is given in [Chapter B](#) of this user manual.

To run an SQL Link macro use the following syntax:

```
Application.run "sqllink.xla!your_macro_name"
```

Here is a sample macro:

```
Sub macro1  
    Range("C3").Select  
    ActiveCell.FormulaR1C1 = "select * from emp"  
    Application.Run "sqllink.xla!Default_Connect"  
    Application.Run "sqllink.xla!Execute_From_Cell"  
    Range("F7:F21").Select  
    Selection.Font.Italic = True  
    Range("F12").Select  
End Sub
```

This macro inserts an sql select statement in a cell, connects to database, executes this sql statement, selects the mgr column and changes its font to italics.

### 3. Using Sendkeys action to execute SQL Link commands

Currently SQL Link menu commands cannot be recorded as part of macro recording. You can get around this problem by using Sendkeys method in your macros.

Here is the slightly modified above macro using Sendkeys method.

```
Sub macro1
    Range("C3").Select
    ActiveCell.FormulaR1C1 = "select * from emp"
    SendKeys "%SCC" ' Connect default user
    SendKeys "%SES" ' Execute SQL in Cell
End Sub
```

Using Sendkeys is not the preferred way since future versions of SQL Link might change the shortcut keys for menu commands. Use direct calls to SQL Link macros whenever possible.

Here is some more description and help for Sendkeys. (This is taken from VBA for XLA help file.)

To specify characters that aren't displayed when you press the corresponding key (for example, ENTER or TAB), use the codes listed in the following table. Each code in the table represents one key on the keyboard.

Key	Code
BACKSPACE	{BACKSPACE} or {BS}
BREAK	{BREAK}
CAPS LOCK	{CAPSLOCK}
CLEAR	{CLEAR}
DELETE or DEL	{DELETE} or {DEL}
DOWN ARROW	{DOWN}
END	{END}
ENTER (numeric keypad)	{ENTER}
ENTER	~ (tilde)
ESC	{ESCAPE} or {ESC}
HELP	{HELP}
HOME	{HOME}
INS	{INSERT}
LEFT ARROW	{LEFT}

NUM LOCK	{NUMLOCK}
PAGE DOWN	{PGDN}
PAGE UP	{PGUP}
RETURN	{RETURN}
RIGHT ARROW	{RIGHT}
SCROLL LOCK	{SCROLLLOCK}
TAB	{TAB}
UP ARROW	{UP}
F1 through F15	{F1} through {F15}

You can also specify keys combined with SHIFT and/or CTRL and/or ALT. To specify a key combined with another key or keys, use the following table.

To combine a key with	Precede the key code with
SHIFT	+ (plus sign)
CTRL	^ (caret)
ALT	% (percent sign)

## A. ODBC Interface

### 1. Executing SELECT Queries

A9		= EASTC				
	A	B	C	D	E	
3	SELECT customerid, companyname, contactname from customers where city = ?					
4						
5	<b>customerid</b>	<b>companyname</b>	<b>contactname</b>			
6	AROUT	Around the Horn	Thomas Hardy			
7	BSBEV	B's Beverages	Victoria Ashworth			
8	CONSH	Consolidated Holdings	Elizabeth Brown			
9	EASTC	Eastern Connection	Ann Devon			
10	NORTS	North/South	Simon Crowther			
11	SEVES	Seven Seas Imports	Hari Kumar			
12						
13	<div><div>Enter Parameter Value</div><div>Please Enter City Name</div><div><input type="text"/></div><div><input type="checkbox"/> Use this value/reference for future refreshes</div><div><div>OK</div><div>Cancel</div></div></div>					
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						

## 2. Specifying User Parameters

## 3. Specifying Other Query Options

	A	B	C	D	E
3	SELECT customerid, companyname, contactname from customers where cit				
4					
5	<b>customerid</b>	<b>companyname</b>	<b>contactname</b>		
6	AROUT	Around the Horn	Thomas Hardy		

**External Data Range Properties** ? x

Name: CUSTOMERS

Query definition

- ☒ Save query definition
- ☒ Save password

Refresh control

- ☒ Enable background refresh
- ☐ Refresh data on file open
- ☐ Remove external data from worksheet before saving

Data layout

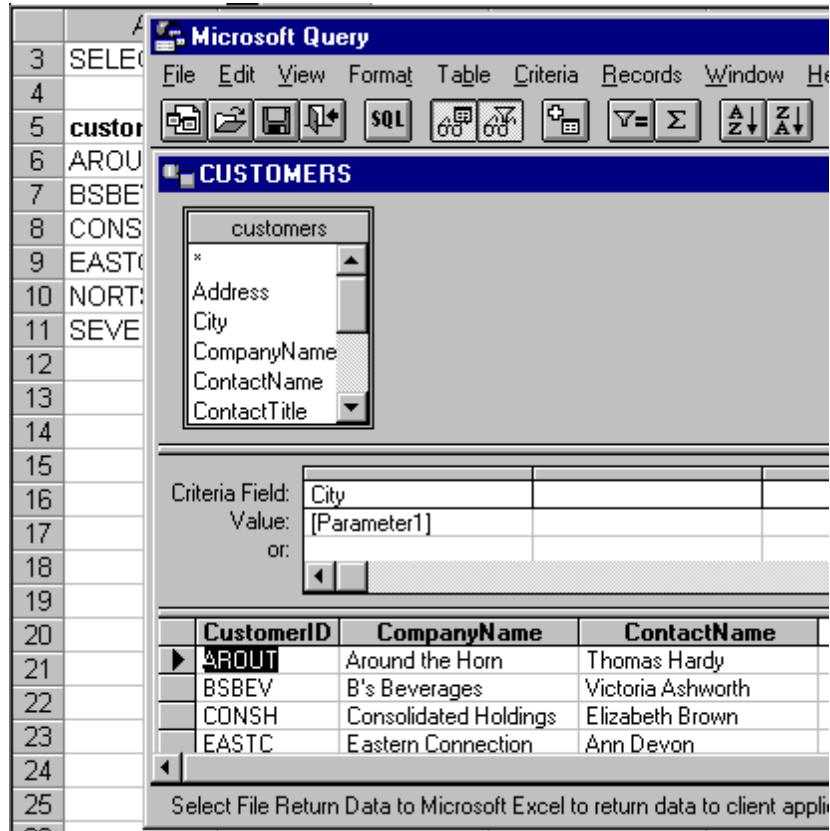
- ☒ Include field names
- ☒ Autoformat data
- ☐ Include row numbers
- ☒ Import HTML table(s) only

If the number of rows in the data range changes upon refresh:

- ☒ Insert cells for new data, delete unused cells
- ☐ Insert entire rows for new data, clear unused cells
- ☐ Overwrite existing cells with new data, clear unused cells



#### 4. Modifying Queries using MS Query Builder



#### 5. Changing SQL created by Query Builder

#### 6. Limitations of ODBC Interface in SQL Link 2.0

## **B. SQL Link Menu and Macros Reference**

### **7. Execute**

#### **SQL in Cell**

Executes SQL in the currently selected cell and paste the result two rows below. Pasted data is also assigned an XL range name using first column in SQL statement as name. You can use “&” variables to specify parameters in the SQL statement.

Macro Name: sqllink.xla!Execute\_From\_Cell

#### **Count Rows**

Counts the rows that will be returned by executing SQL in the currently selected cell and pastes this count in two rows below.

Macro Name: sqllink.xla!Count\_Rows

#### **SQL in Comment – One**

Executes SQL in the comment text of currently selected cell and pastes the result in this column and below.

Macro Name: sqllink.xla!Exe\_Comment\_One

#### **SQL in Comment – Selection**

Same as above but executes SQL from comment text for the currently selected cells one-by-one.

Macro Name: sqllink.xla!Exe\_Comment\_Sel

#### **Batch Execute One**

Executes currently selected SQL statement from SQL\_Batch sheet. Cursor must be positioned in the first column of the row in SQL\_Batch sheet that contains the SQL to be executed.

Macro Name: sqllink.xla!Execute\_One

#### **Batch Execute All**

Automatically switches to “sql\_batch” sheets and executes SQL statements one-by-one. After execution it returns to the original sheet. Cursor can be anywhere in any sheet.

Macro Name: sqllink.xls!Execute\_All

## **DB/FS Script**

### **8. Connect**

#### **Connect Default User**

Connects to the database using user name, password and database name specified on “Options” sheet.

Macro Name: sqllink.xls!Default\_Connect

#### **Connect User**

Prompts the user for user name, password, database and connects to database.

Macro Name: sqllink.xls!User\_Connect

### **9. Dictionary**

#### **List Tables**

Lists the tables of current users in the currently selected cell and below.

Macro Name: sqllink.xls!List\_Tables

#### **List Columns**

Lists the column names of the table whose name is under the cursor.

Macro Name: sqllink.xls!List\_Columns

#### **Table Quick View**

Displays the first 10 rows of the table under selection in a new sheet.

Macro Name: sqllink.xls!Quick\_View

## 10. Query by Example

### Build QBE Grid

This command creates and formats an area of current sheet which you can use to do Query-by-Example. To execute this command.

1. Enter the table name in a worksheet cell.
2. Position the cursor on this cell and give the command SQL : Query by Example : Build QBE Grid.

Macro Name: sqllink.xla!Build\_QBE

### Execute QBE

This command builds the SQL from the criteria entered in QBE grid and executes this SQL. To give this command, you must position the cursor in the first column of the QBE grid.

Macro Name: sqllink.xla!Execute\_QBE

## 11. Data

### PL/SQL in Cell

Executes PL/SQL (anything except SELECT statement) in the currently selected cells and pastes the number of rows affected two lines below.

Macro Name: sqllink.xla!Execute\_PLSQL

### Create Insert Rows Area

Using the SQL statement in current cell, it creates an area where you can enter data.

Macro Name: sqllink.xla!Insert\_Data1

### Insert Rows

Inserts the data that you have entered using the above-mentioned command.

To execute this command, select the cell with SQL statement (used to create insert rows area) and give this command from OracleSQL menu.

Each INSERT or UPDATE operation starts a new session or participates in the current session. Data is committed to database only when you give the COMMIT command on Data menu.

You can also ROLLBACK your work by ROLLBACK command on the Data menu.

Macro Name: sqllink.xls!Insert\_Data2

## Update Rows

Updates the data retrieved using SQL select command.

To execute this command, select the cell with SQL statement (which was executed to retrieve this data) and give this command from OracleSQL menu.

Each INSERT or UPDATE operation starts a new session or participates in the current session. Data is committed to database only when you give the COMMIT command on Data menu.

**WARNING USE WITH CARE:** This command currently does not use primary key or any such mechanism to update data. It just updates the last retrieved data. This might not be suitable in certain situations (e.g. somebody has inserted new rows in the data you retrieved) and you may get unpredictable results.

Macro Name: sqllink.xls!Update\_Data

## Commit Session

Commits the currently active session to database and makes any changes permanent made through INSERT or UPDATE etc.

Macro Name: sqllink.xls!Commit\_Session

## Rollback Session

Rollbacks the currently active session and discards any changes made to the database in current session.

Macro Name: sqllink.xls!Rollback\_Session

## 12. Generate

### Select

Generates a simple SELECT statement in the current cell assuming the contents of current cell as table name.

Macro Name: sqllink.xls!Gen\_Select

### Insert

Generates a simple SELECT statement in the current cell assuming the contents of current cell as table name.

Macro Name: sqllink.xls!Gen\_Insert

### Update

Generates a simple SELECT statement in the current cell assuming the contents of current cell as table name.

Macro Name: sqllink.xls!Gen\_Update

### Delete

Generates a simple SELECT statement in the current cell assuming the contents of current cell as table name.

Macro Name: sqllink.xls!Gen\_Delete

## 13. Key Builders

### Numbers 2 Text

Converts the numbers in the current selection to text and optionally pads with zero to the length which is specified in the Text Processing Section of Options sheet.

Macro Name: sqllink.xls!Num2Text

### Text 2 Numbers

Converts the text in the current selection to numbers. If something cannot be translated in a valid number, it is converted to use.

Macro Name: sqllink.xls!Text2Num

### **Concatenate Columns**

Concatenates the columns in current selection using the concatenation separator specified on Options sheet. It then puts the results in the first column of the selection, overriding any existing data.

Macro Name: sqllink.xls!Concat\_Cols

## **14. Report Builders**

### **SUM Selection**

Inserts a SUM formula for the currently selected cell ranges.

Macro Name: sqllink.xls!Sum\_Selection

### **Named Range Lookup**

Inserts a VLOOKUP formula using the criteria given by the user.

Macro Name: sqllink.xls!Lookup\_Wiz2

### **Selected Range Lookup**

Macro Name: sqllink.xls!Lookup\_Wizard

### **Change Numbers**

Macro Name: sqllink.xls!Change\_Numbers

## **15. Oracle Link**

### **Insert Options Sheet**

Macro Name: sqllink.xls!Insert\_SQLLink\_Sheets

### **Auto\_Open Macro**

Macro Name: sqllink.xls!CreateAnAutoOpen

## 16. ODBC

### Execute SQL

Macro Name: sqllink.xla!ODBC\_Execute\_SQL

### Get SQL

Macro Name: sqllink.xla!ODBC\_Get\_SQL

## 17. About

### About

Macro Name: sqllink.xla!Show\_About



## **C. Trouble Shooting and Technical Support**

## **D. License and Disclaimer**

**SQL LINK 1.50 IS A FREE SOFTWARE WITHOUT ANY WARRANTIES AND SUPPORT.**

### **Distribution**

You can distribute and make as many copies of this software and documentation as you wish. You can also give exact copies of the original version to anyone; and distribute the software and documentation in its unmodified form via electronic means. There is no charge for any of the above.

You are specifically prohibited from charging, or requesting donations, for any such copies, however made; and from distributing the software and/or documentation with other products (commercial or otherwise) without prior written permission.

### **Disclaimer of Warranty**

THIS SOFTWARE AND THE ACCOMPANYING FILES ARE SOLD "AS IS" AND WITHOUT WARRANTIES AS TO PERFORMANCE OR MERCHANTABILITY OR ANY OTHER WARRANTIES WHETHER EXPRESSED OR IMPLIED. NO WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE IS OFFERED. ANY LIABILITY OF THE SELLER WILL BE LIMITED EXCLUSIVELY TO PRODUCT REPLACEMENT OR REFUND OF PURCHASE PRICE.

**It is a good practice to thoroughly test *SQL Link* with non-critical data before relying on it. The user must assume the entire risk of using the program.**

**By using SQL Link, user agrees to the above terms and conditions of disclaimer of warranty.**

**If you do not agree uninstall and delete ALL sql link files from your workstation/server/network immediately.**